26. The method of distributed cryptographic computation as recited by claim 1 wherein the cryptographic computation comprises digital signing.

27. The method of distributed cryptographic computation as recited by claim 12 wherein the cryptographic computation comprises digital signing.

28. The method of distributed cryptographic computation as recited by claim 1 further comprising a step of using generated random values or shared values to detect misbehaving devices.

29. The method of distributed cryptographic computation as recited by claim 12 further comprising a step of using generated random values or shared values to detect misbehaving devices.

30. The method of distributed cryptographic computation as recited by claim 1 further comprising a step of proactively updating a secret cryptographic value used in the cryptographic computation.

31. The method of distributed cryptographic computation as recited by claim 12 further comprising a step of proactively updating a secret cryptographic value used in the cryptographic computation.

REMARKS

The applicant respectfully requests reconsideration and allowance of this application. After entry of this Amendment, claims 1-12, 17-31 will be pending, with claims 1-12 and 17-22 amended, claims 13-16 deleted, and claims 23-31 newly submitted for examination.

Claims 1-10 were rejected as being anticipated by Gennaro et al. Claims 1-4 and 9-22 were rejected as being anticipated by Brickell et al. The Applicant respectfully submits that the

distinctions and merits of this application will be better appreciated with reference to an embodiment disclosed in pages 9 - 12 of the specification, which may be implemented in an architecture shown in Fig. 1. For clarity in communicating concepts, shared randomness will be discussed here with reference to a specific example of distributed signing. Such specificity in explanation is not intended to limit the scope of protection.

Fig. 1 shows an architecture having five signing units. Those devices may be designated as "1," "2," "3," "4," and "5." During a setup phase, members of the system adopt a series of pseudorandom functions $PRF_k(\cdot)$ indexed by variable "k." During later phases, the variable "k" may take on specific values depending on context. Also during the setup phase, each pair of signing devices jointly generates a shared secret key $\sigma_{i,j}$. For example, signing devices "1" and "2" both generate the same shared key $\sigma_{1,2}$ (which is identical to $\sigma_{2,1}$), signing devices "1" and "3" generate a different shared key $\sigma_{1,3}$ and so on for all pairs that can be formed among the five devices. The values of σ may be used as the index value "k" for the pseudorandom function.

The σ values are "shared" in the sense that each device fully possess each shared value. "Sharing" in this sense is different from, e.g., Shamir secret-sharing where each of multiple "shareholders" receives a "share" or number related to a secret, but does not have sufficient information to obtain the secret. While other cryptographic values, such as a signing key, may be "shared" in the sense that a device has number related to a secret without sufficient information to obtain the secret, the σ values are fully known to the entities that share them.

During an operation to sign a specific message m, a subset of devices may be selected, such as three out of the five. If the three devices are "1," "2," and "3," the set $\Lambda = \{1, 2, 3\}$. As described on page 11 of the specification, each member of the set Λ will compute a value $s'_{m,j,\Lambda}$ that includes the following term: $\sum v \epsilon \Lambda / j \ \text{sign}(j - v) \bullet PRF\sigma_{j,v}$. In that term, the sum is taken over

all devices v that are elements of the set Λ excluding device j. For signing device "1," j = 1, and the summation has two terms: a term for v = 2 and a term for v = 3. The terms of the summation would be:

$$\text{sign}(1 - 2) \bullet \text{PRF } \sigma_{1,2}(m) + \text{sign}(1 - 3) \bullet \text{PRF } \sigma_{1,3}.(m).$$

Signing device "2" would have terms for j = 2 and v = 1 , 3. Signing device "3" would have terms for j = 3 and v = 1, 2.

Each pair of signing devices shares a source of randomness in the form of the function PRF $\sigma_{j,v}$. Because $\sigma_{j,v} = \sigma_{v,j}$ (they are the same number, which both members know), pair members will select the same pseudorandom function ($\text{PRF}\sigma_{j,v} = \text{PRF}\sigma_{v,j}$) and contribute partial results with "random" contributions that are actually related to one another. An error or misbehavior of a participant will be revealed if contributions do not relate properly. For example, a final signature will not verify unless both members of a pair contribute shared random values with the required relationship. As stated on page 10 of the specification, the common knowledge of the pseudorandom functions and their invocation in the computation generates a "t-wise hand shake."

Neither Gennaro et al. nor Brickell et. al. disclose such a sharing of sources of randomness for use in computing. (They may "share" some cryptographic values in the sense that group members each may hold a number that is related to a secret, but no one member has sufficient information to obtain the secret.)

Claims 1 and 12 have been amended to clarify that shared values are shared among distinct subsets of devices, which in the example discussed above would be a pair-wise sharing. Other claims have been amended for internal consistency and clarity, but not for the purpose of establishing patentability.

The Examiner stated a marked-up copy of claim 1 with markings to show changes was not submitted with the communication filed on April 2, 2001. A marked-up copy of claim 1 as amended in such communication is attached.

In light of the above remarks, it is believed the merit of this application will be appreciated and that the application will be passed to issuance. If, however, the Examiner is not persuaded, the applicant requests an opportunity for a personal interview at the Examiner's earliest convenience.

Respectfully submitted,

by J. Douglas Finnie Reg. No. 24,514

Date: December 31, 2001

Stuart T. F. Huang
Registration No. 34,184
Steptoe & Johnson, LLP
1330 Connecticut Avenue, N.W.
Washington, DC 20036
Tel: (202) 429-8056;
Fax: (202) 429-3902

VERSION WITH MARKINGS TO SHOW CHANGES

1. A method of distributed cryptographic computation using a plurality of distributed electronic devices, said method comprising:

(a) computing shared values over a known and agreed context, each value being shared among a distinct subset of the plurality of distributed electronic devices;

(b) at each of a plurality of the distributed electronic devices, generating a random value [values] using said shared values;

(c) at each of a plurality of the distributed electronic devices, generating a partial result for the distributed cryptographic computation using at least one of said random values; and

(d) computing a final result for the distributed cryptographic computation using partial results.

2. The method of [using] distributed cryptographic [keys] computation as recited by claim 1, wherein said shared values are random keys.

3. The method of [using] distributed cryptographic [keys] computation as recited by claim 1, wherein said shared values are derived from a cryptographic protocol.

4. The method of [using] distributed cryptographic keys as recited by claim 1, wherein said shared values are derived cryptographically.

5. The method of [using] distributed cryptographic [keys] computation as recited by claim 1, further comprising the step of implementing a re-representation of a function.

6. The method of [using] distributed cryptographic [keys] computation as recited by claim 1, wherein said partial results may include incorrect values.

7. The method of [using] distributed cryptographic [keys] computation as recited by claim 1, wherein said steps (a)-(d) are performed iteratively.

8. The method of [using] distributed cryptographic [keys] computation as recited by claim 7, further comprising changing said shared values after said step of generating an output based on said partial result.

9. The method of [using] distributed cryptographic [keys] computation as recited by claim 3, wherein said cryptographic protocol is a cryptographic function involving exponentiation.

10. The method of [using] distributed cryptographic [keys] computation as recited by claim 3, wherein said cryptographic protocol is an RSA function.

11. The method of [using] distributed cryptographic [keys] computation as recited by claim 1, wherein said shared values are stored in a hardware device in at least one of said distributed electronic devices.

12. A method of distributed cryptographic computation using a cryptographic value shared among a plurality of distributed electronic devices, said method comprising:

(a) selecting a subgroup of devices to perform the distributed cryptographic communication

(b) computing shared values over a known and agreed context, each value being shared among a distinct subset of the subgroup of distributed electronic devices;

[(b)] (c) at each distributed electronic device of the subgroup, generating a random value [values] using said shared values;

[(c)] (d) at each device of the subgroup [of a plurality] of [the] distributed electronic devices, generating a partial result for the cryptographic computation using a share of the cryptographic value and at least one of said random values; and

[(d)] (e) computing a final result for the distributed cryptographic computation using partial results.

17. The method of distributed cryptographic computation as recited by claim 1, wherein each of [a plurality of] the computed, shared values is shared among a pair of the distributed electronic devices.

18. The method of distributed cryptographic computation as recited by claim 12, wherein each of [a plurality of] the computed, shared values is shared among a pair of the distributed electronic devices.

19. The method of distributed cryptographic computation as recited by claim 1, wherein each [of a plurality of shared values] computed, shared value is shared among a distinct pair of the distributed electronic devices.

20. The method of distributed cryptographic computation as recited by claim 12, wherein each [of a plurality of shared values] computed, shared value is shared among a distinct pair of the distributed electronic devices.

21. The method of distributed cryptographic computation as recited by claim 1, wherein each [of a plurality of shared values] computed, shared value is (a) shared among a distinct subset of distributed electronic devices and (b) used to generate a partial result in a way that permits verification of correctness of a partial result.

22. The method of distributed cryptographic computation as recited by claim 12, wherein each [of a plurality of shared values] computed, shared value is (a) shared among a distinct subset of distributed electronic devices and (b) used to generate a partial result in a way that permits verification of correctness of a partial result.

--23. (New) The method of distributed cryptographic computation as recited by claim 12 wherein the random values depend upon the particular set of devices selected for the subgroup.

24. (New) The method of distributed cryptographic computation as recited by claim 1 wherein the cryptographic computation is based on an argument, and the generated random values are based on said argument.

25. (New) The method of distributed cryptographic computation as recited by claim 12 wherein the cryptographic computation is based on an argument, and the generated random values are based on said argument.

26. (New) The method of distributed cryptographic computation as recited by claim 1 wherein the cryptographic computation comprises digital signing.

27. (New) The method of distributed cryptographic computation as recited by claim 12 wherein the cryptographic computation comprises digital signing.

28. (New) The method of distributed cryptographic computation as recited by claim 1 further comprising a step of using generated random values or shared values to detect misbehaving devices.

29. (New) The method of distributed cryptographic computation as recited by claim 12 further comprising a step of using generated random values or shared values to detect misbehaving devices.

30. (New) The method of distributed cryptographic computation as recited by claim 1 further comprising a step of proactively updating a secret cryptographic value used in the cryptographic computation.

31. (New) The method of distributed cryptographic computation as recited by claim 12 further comprising a step of proactively updating a secret cryptographic value used in the cryptographic computation.--

VERSION WITH MARKINGS TO SHOW CHANGES

1. A method of distributed cryptographic computation ~~between~~ using a plurality of distributed electronic devices, ~~said distributed electronic devices capable of communication with a central server,~~ said method comprising ~~the steps of~~:

(a) computing shared values over a known and agreed context;

(b) generating random values using said shared values;

(c) at each of a plurality of the distributed electronic devices, generating a partial result for the distributed cryptographic computation ~~each device~~ using at least one of said random values; and

(d) computing ~~an output based on said partial result~~ a final result for the distributed cryptographic computation using partial results.